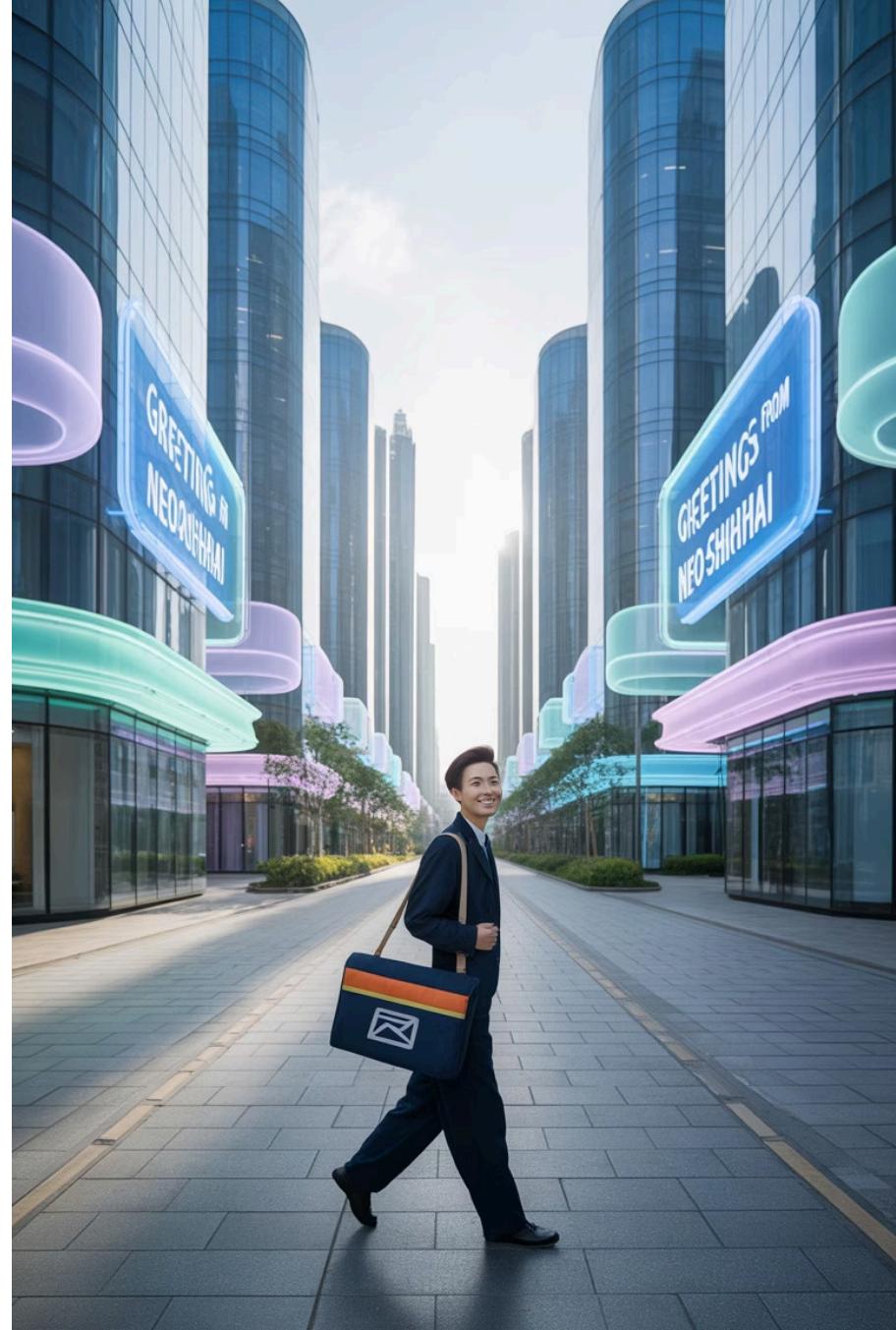


Problema do Carteiro Chinês (PCC)

Projeto e Análise de Algoritmos — Seminário

Apresentadores: Gilson Inácio da Silva · Ederson Manoel de Oliveira



Problema



Coleta de Lixo

Caminhões de lixo devem cobrir todas as ruas minimizando combustível e tempo.

Do Bairro ao Grafo

Definição Matemática

Grafo não dirigido e conexo: $G = (V, E, w)$
com $w(e) \geq 0$

- **Vértices:** representam esquinas
- **Arestas:** representam ruas
- **Peso:** distância ou comprimento

Casos Possíveis

Grafo Euleriano: Todos os vértices têm grau par → circuito de Euler resolve diretamente

Vértices Ímpares: Precisamos "eulerizar" duplicando trechos mínimos



Ideia do Algoritmo



Encontrar Vértices Ímpares

$O = \{v \in V \mid \deg(v) \text{ ímpar}\}$



Calcular Distâncias Mínimas

$d(u,v)$ entre todos os pares $u,v \in O$ usando Dijkstra



Pareamento Perfeito Mínimo

Resolver em O usando programação dinâmica por bitmask



Duplicar Arestas

Duplicar arestas dos caminhos mínimos escolhidos

Análise de Complexidade

Cálculo de Distâncias

$$O(|O| \cdot |E| \log |V|)$$

Dijkstra executado para cada vértice ímpar como origem

Pareamento (Gargalo)

$$O(|O|^2 \cdot 2^{|O|})$$

Programação dinâmica por bitmask - crítico quando $|O|$ é grande

Algoritmo de Hierholzer

$$O(|E|)$$

Extração do circuito euleriano é linear nas arestas

Na prática urbana: $|O|$ costuma ser moderado, permitindo que a solução execute rapidamente em casos reais.

Fórmula do Custo Ótimo

$$C^* = \sum_{e \in E} w(e) + \min_{M \text{ matching em } O} \sum_{(u,v) \in M} d(u, v)$$

Primeiro Termo

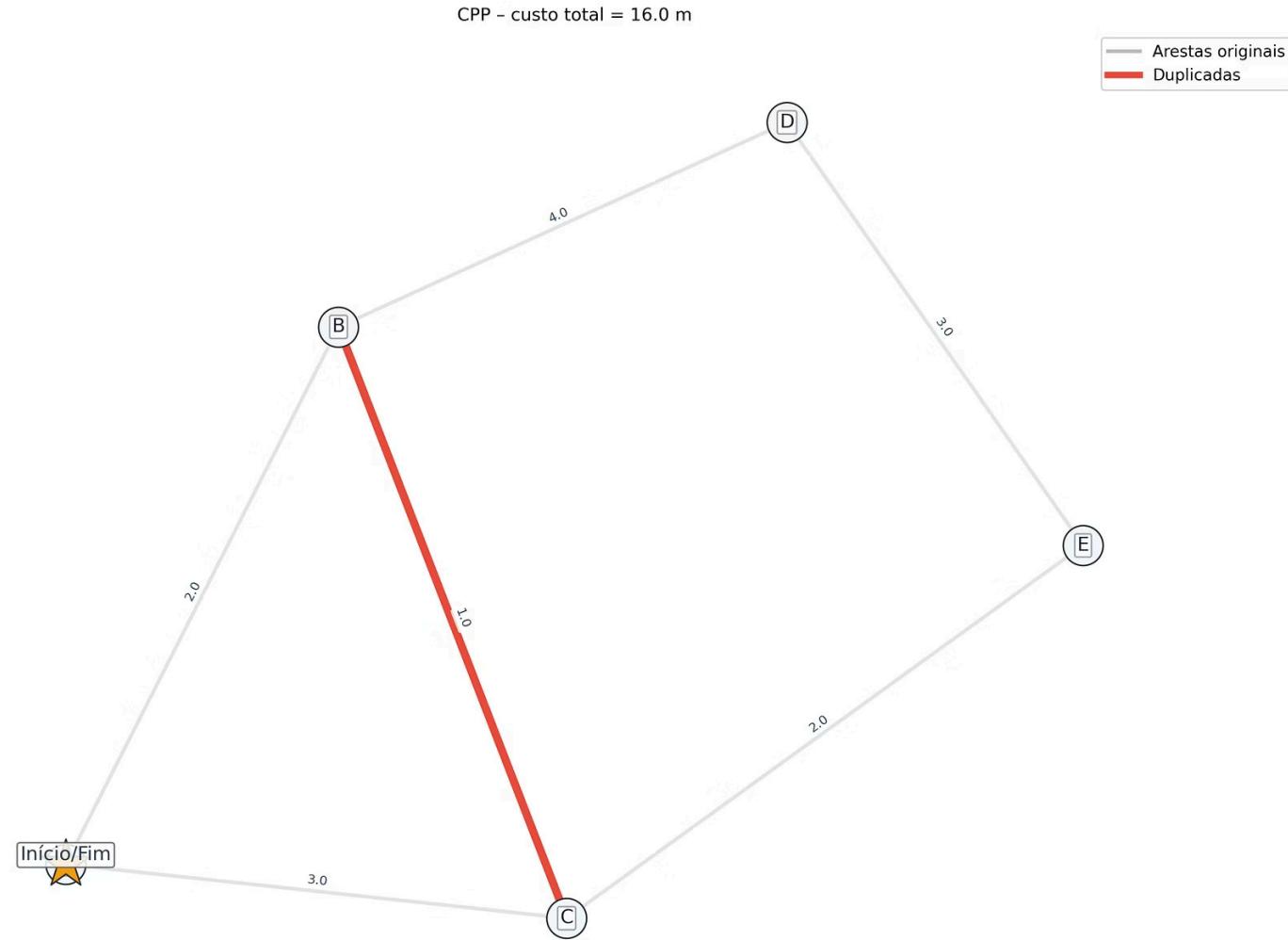
Soma dos pesos de todas as arestas originais do grafo

Segundo Termo

Custo mínimo do pareamento perfeito entre vértices ímpares

 **Resultado:** Esta fórmula garante a solução ótima para o Problema do Carteiro Chinês em grafos não dirigidos.

Exemplo Didático (A-E)



Configuração do Problema

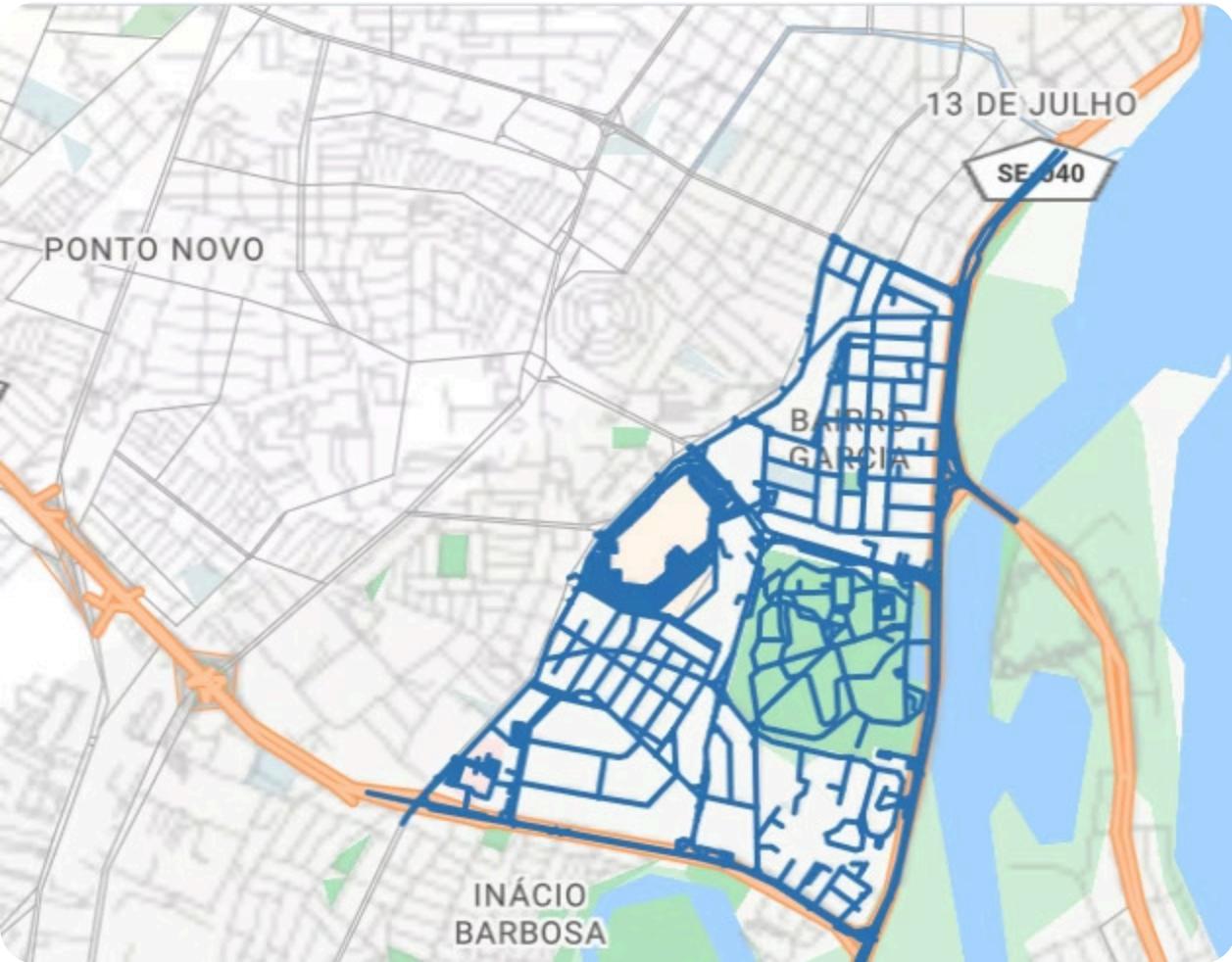
- Instância em `data/example_edges.csv`
- Vértices ímpares: B e C
- Solução: duplicar aresta B-C (peso 1)
- Resultado: todos os vértices ficam pares

Custo total: $15 + 1 = 16$

Interpretação Visual

- Cinza:** ruas originais do grafo
Vermelho: trechos duplicados para completar o circuito
Estrela: ponto de início e fim do percurso

Caso Real com Mapa



Visualização do **Bairro Jardins, Aracaju, SE** com mapa base do OpenStreetMap, mostrando o contexto geográfico real da solução do Problema do Carteiro Chinês.

- Esta versão inclui o mapa base para melhor contextualização geográfica
- Permite identificar ruas, quarteirões e pontos de referência do bairro
- Útil para apresentações que requerem localização precisa
- Requer conexão com internet durante a geração (comando --basemap)

Caso Real: Bairro Jardins

Dados do OpenStreetMap convertidos de GeoJSON para formato CSV com coordenadas u,v,w e informações dos nós id,lat,lon.

6.0

Custo Aproximado

Quilômetros de percurso total (varia conforme recorte da área)

6.3

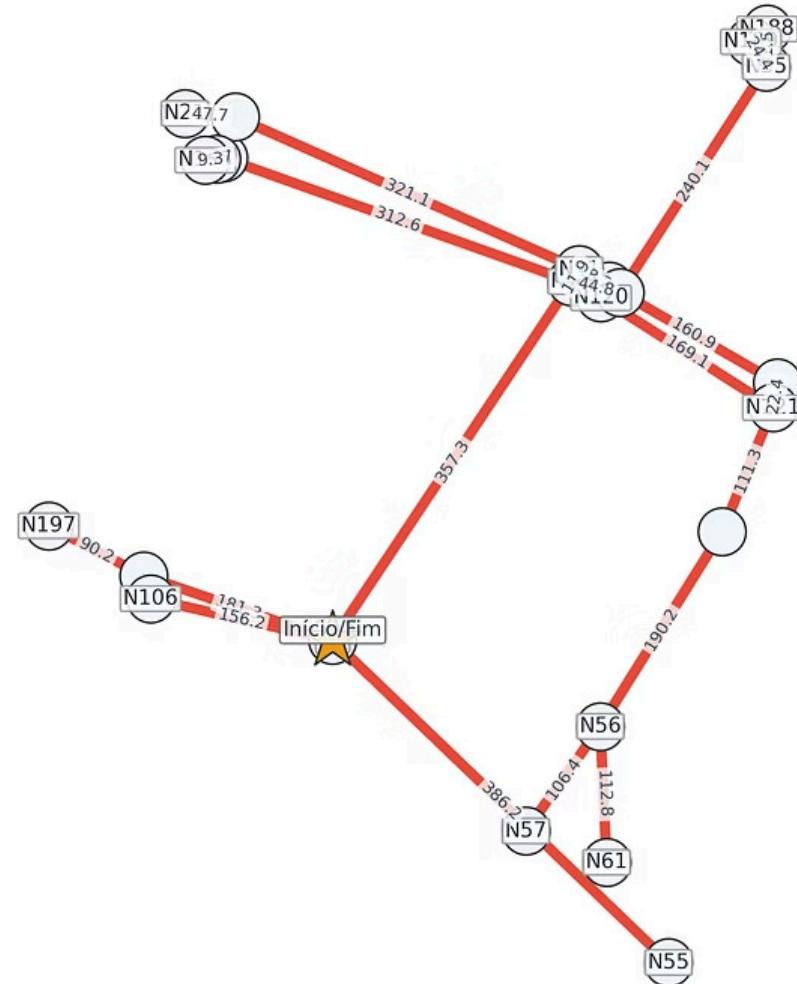
Variação Máxima

Quilômetros dependendo dos limites do bairro considerado

Legenda da Visualização

- **Cinza:** rede viária original
- **Vermelho:** duplicações mínimas (becos e ramos)
- **Estrela:** marca o ponto de início e fim

Crédito: Map data © OpenStreetMap contributors, ODbL 1.0



Limitações e Desafios



Gargalo Principal

Matching por DP explode exponencialmente quando há muitos vértices ímpares



Grafos Dirigidos

Versões dirigidas exigem balanceamento de grau por algoritmos de fluxo



Dependência de Internet

Basemap requer conexão; para apresentações, prefira versão sem mapa

Alternativas e Melhorias

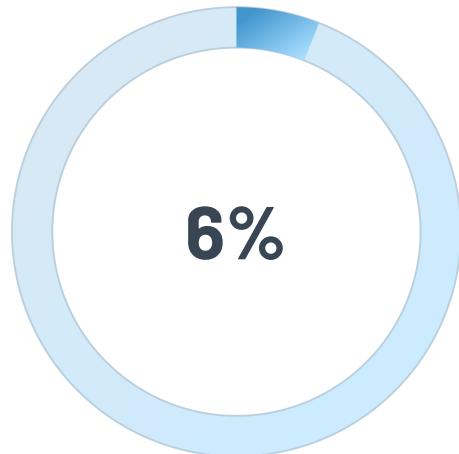
- **Algoritmos alternativos:** Blossom/Edmonds para matching
- **Heurísticas:** para instâncias gigantes onde solução ótima é impraticável

Resultados Obtidos



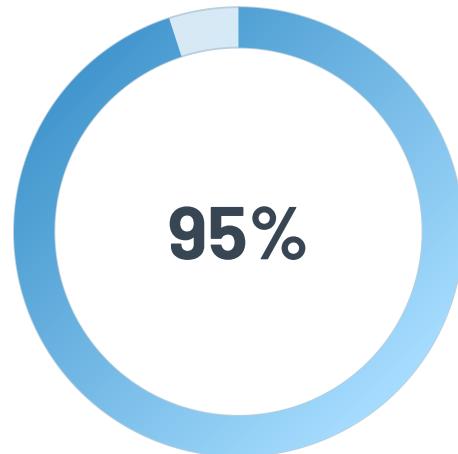
Cobertura Garantida

Todas as ruas são percorridas pelo menos uma vez



Overhead Típico

Duplicações necessárias em relação ao percurso total



Eficiência Alcançada

Redução significativa comparada a abordagens ingênuas

O caso do **Bairro Jardins** demonstra a **aplicabilidade prática**: aproximadamente **6 km de percurso total**, com duplicações mínimas concentradas em becos e ramos da rede viária.

Conclusões Principais

Modelagem Eficaz

O PCC modela com precisão tarefas urbanas de cobertura de ruas, oferecendo solução ótima para grafos não dirigidos

Pipeline Completo

Sistema integrado: dados → grafo → solver → visualização → exportação em múltiplos formatos

Aplicação Prática

Caso do **Bairro Jardins** comprova viabilidade: **~6 km** de percurso com duplicações mínimas estratégicas

O repositório no github disponibiliza comandos prontos, figuras dos exemplos e casos reais, além dos slides completos para replicação e estudo.